End-to-End Problems and Solutions in EOSDIS

UC Berkeley: Michael Stonebraker & James Demmel

UC Santa Barbara: Jeff Dozier & David Siegel

UC San Diego: Joseph Pasquale

UCLA: Roberto Mechoso

DEC/SDSC: Peter Kochevar

LLNL: Richard Watson

http://s2k-ftp.cs.berkeley.edu:8000/nasa_e2e/

Acknowledgements

UC Berkeley:	Paul Brown, Richard Troy, Sharon Smith, Paul Aoki, Keith Sklower, Jeff Sidell
UC Santa Barbara:	Jean Anderson, Debbie Donahue, Jim Frew
UC San Diego:	Norival Figueira
UCLA:	Yuechen Chi
DEC/SDSC:	Len Wanger, Santiago Becerra, Ryan Camoras
LLNL:	Jim McGraw, Steve Louis

Goals

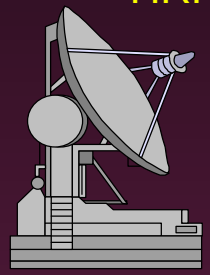
- Validate the UC-proposed *Alternative Architecture for EOSDIS* by building a 1:1,000 prototype
- Explore bottlenecks in architecture by modeling
- Conduct research to plug holes in architecture

Tenets of the architecture

- All data in commercial object-relational DBMS with a common schema
- Manage workflow as user-defined functions
- Track data lineage
- *Eager* or *lazy* evaluation
- Distributed DBMS middleware
- Visualization and science products as DBMS applications

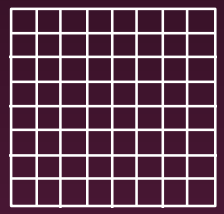
E.g.: AVHRR-TOPEX Fusion

HRPT Data Handling



HRPT downlink

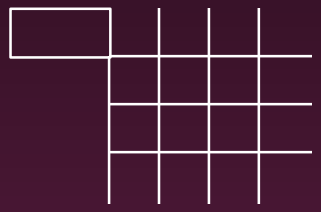
TeraScan →



AVHRR image

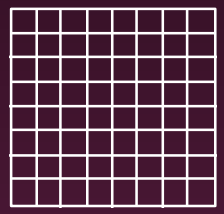
HRPT Gridding

calibrate, de-cloud, aggregate →



SST "super data"

extrapolate →



SST grid

GOAL:
Local and advective components of upper ocean heat balance:

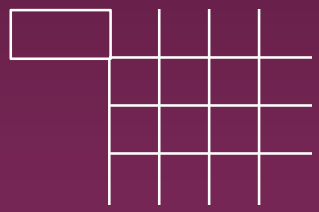
$$\frac{SST}{t} + \vec{u} \cdot \nabla SST$$

TOPEX Data Handling

CD-ROM

MGDR cracker →

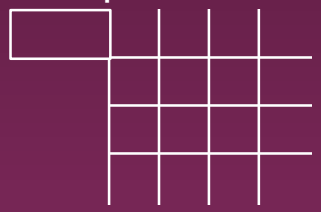
TOPEX points



TOPEX Gridding

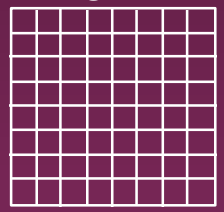
calibrate, aggregate →

ocean height "super data"



extrapolate →

ocean height grid



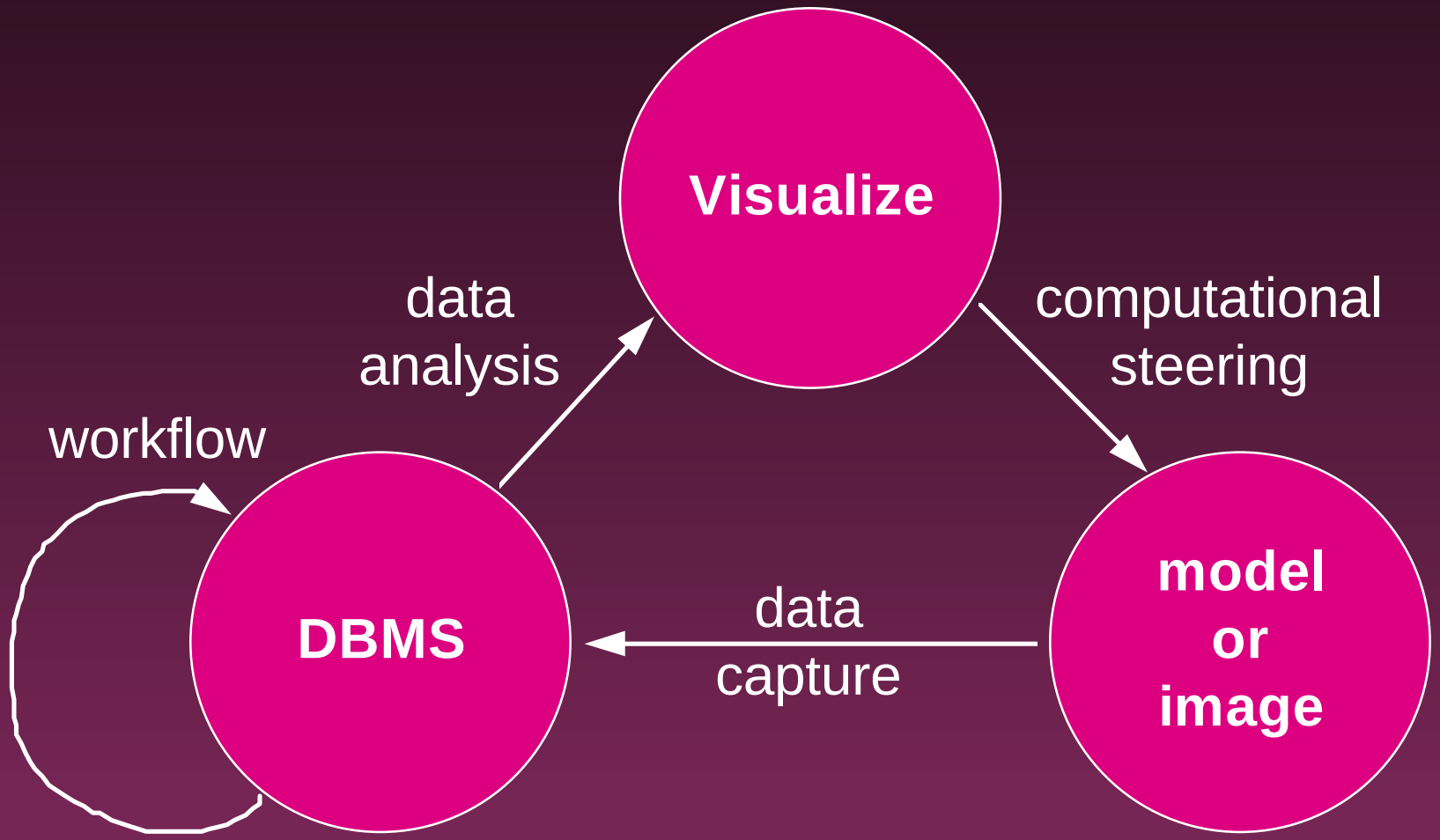
Prototype is in use

- UCSB
  - » Supports satellite data analysis for Dozier and Siegel
- UCLA
  - » Supports modeling of climate, ocean, and atmospheric chemistry for Mechoso et al.
- Supports all tenets of the architecture
  - » except distributed DBMS, in progress
- See the demo!
  - » loading, eager and lazy evaluation
  - » visualization of GCM output
  - » location transparency

Specifics about the prototype

local DBMS:	Illustra
schema:	based on FGDC, SAIF, and lots of work
DBMS type extensions:	required to get anywhere
eager or lazy evaluation:	specified *a priori,* can save new objects
lineage:	part of schema
middleware:	primitive, in Tcl
GUI:	Illustra “Object Knowledge”

Modeling: the Mechoso Triangle



What is the bottleneck?

①CPU cycles for model?

②Network bandwidth for internal model data or DBMS capture?

③I/O or CPU for DBMS or workflow functions?

④I/O or CPU for visualization?

- Today ① is unlikely to be the bottleneck
- ② is the bottleneck if storage poorly arranged

Key technological problems with architecture

DBMS↔tertiary memory interface must be improved

- working with National Storage Lab at LLNL
- to be integrated with our prototype

Key technological problems with architecture

Scalability of distributed DBMS, with network, to ~1,000 DAACs and SCFs

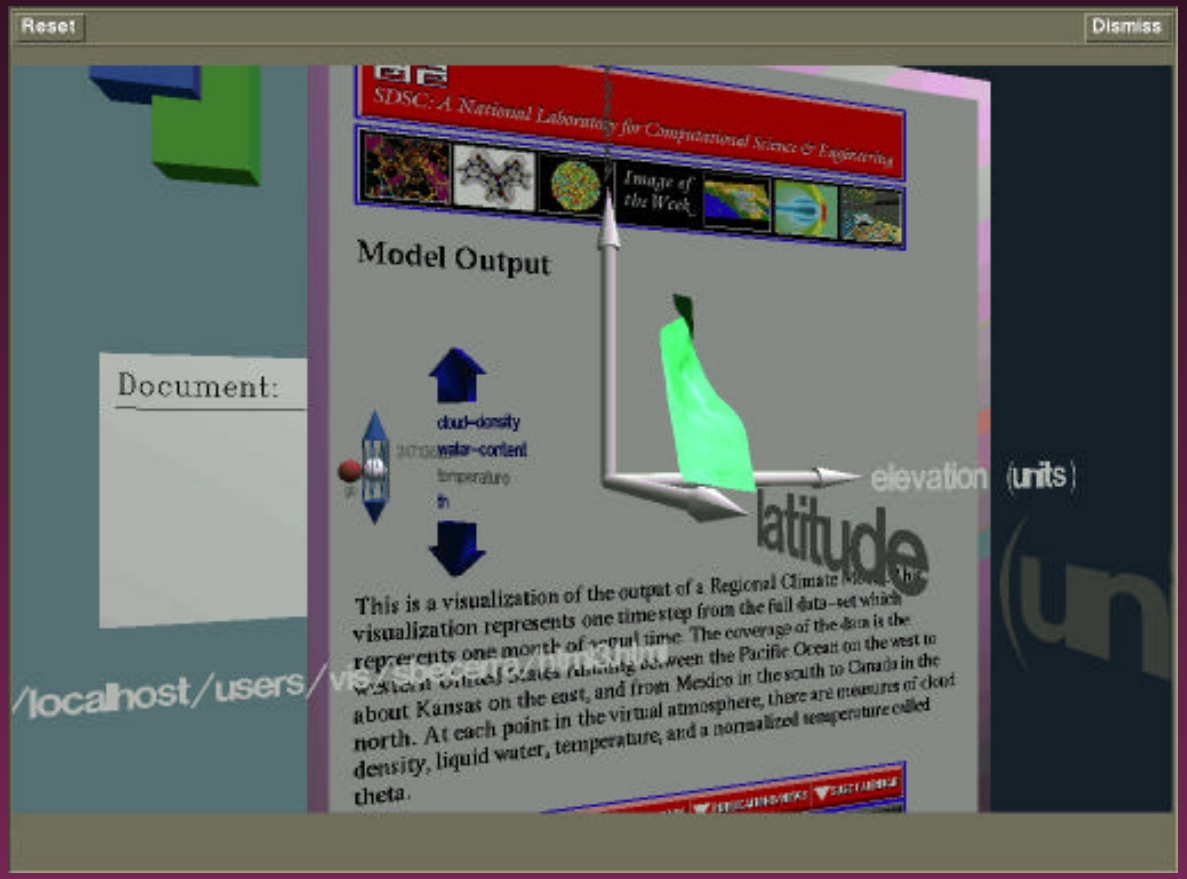
- Mariposa: a scalable, location-transparent, distributed DBMS
- Based on economic paradigm
  - » with network, CPU, and I/O resources
- See the demo!

Key technological problems with architecture

Visualization is too primitive and too hard

- Tecate: general data-exploration utility that leverages
  - » Java/VRML-like interpretative language
  - » 3-D hardware
  - » object-oriented specification
  - » DBMS connection

Tecate example



Plans for distribution

- tar-wad “available” now (to real good friends)
- Robust version in early 1996